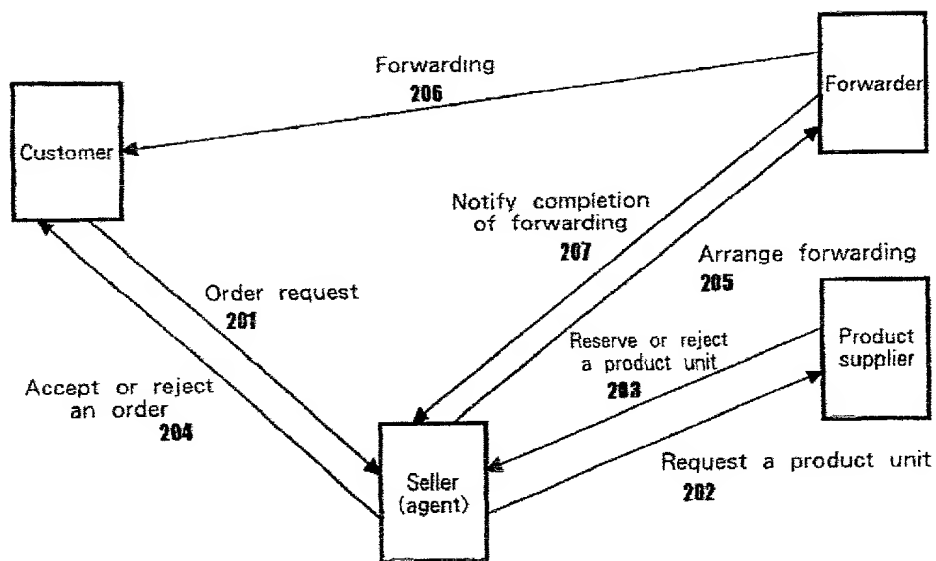
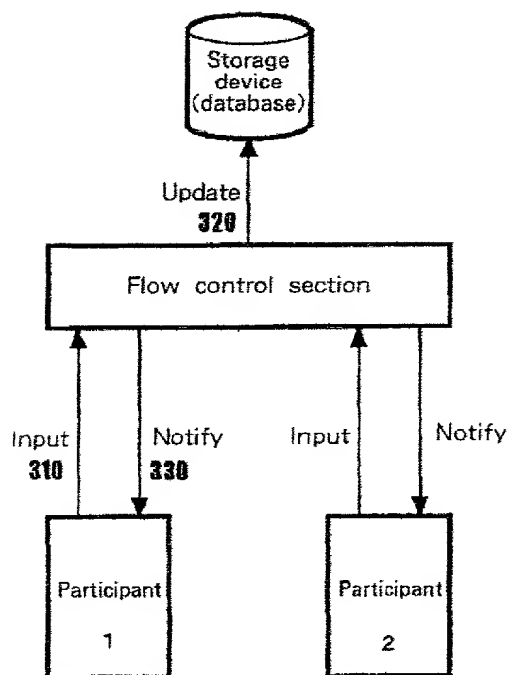


Fig. 1



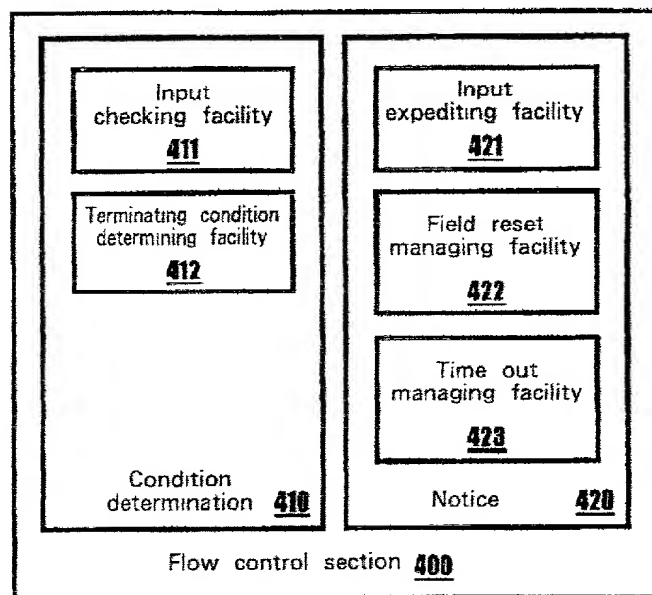
Typical inter - company workflow

Fig. 2



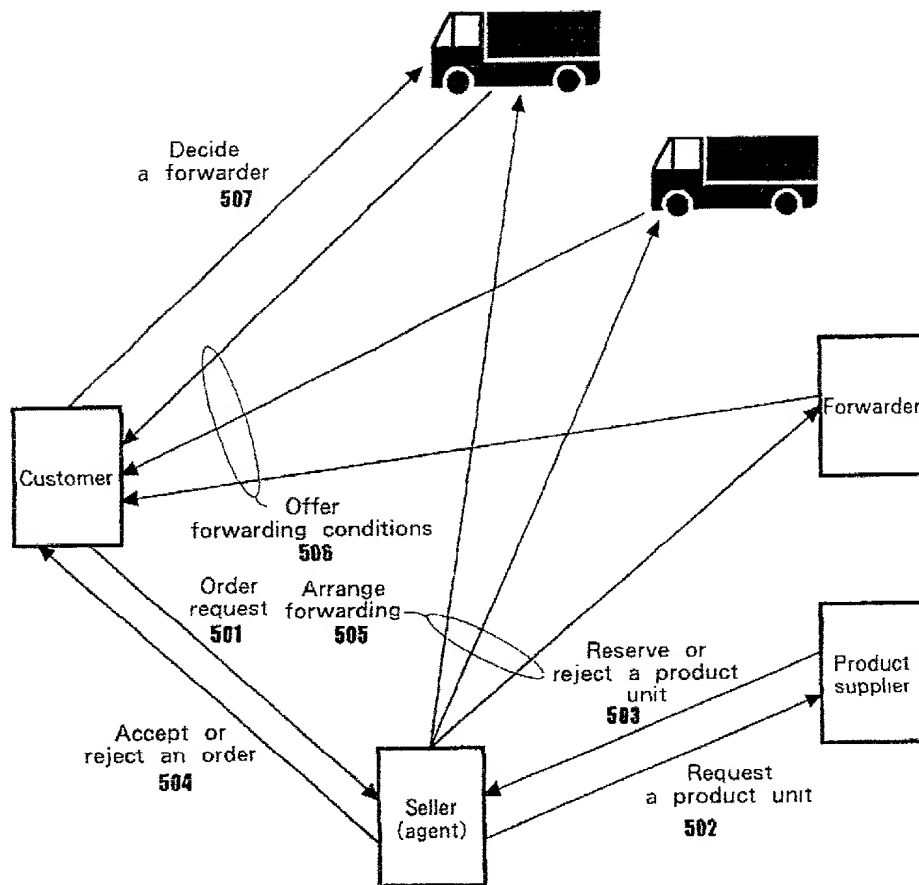
Overview of a workflow controlling system

Fig. 3



Structure of a flow control section

Fig. 4



Workflow including a bid from a forwarder

Fig. 5

Contents
 Tree structure
 (Node, [value])
 History
 (Time, person, action, node ID)
 Access Control
 (Node ID, tag name, person, role, action, conditional expression)
 Constrains
 (Conditional expression)
 Dependencies
 (Depended node ID, Dependent node ID)
 Termination
 (Type, conditional expression)
 Type: End or Abort

[A] means that A is optional

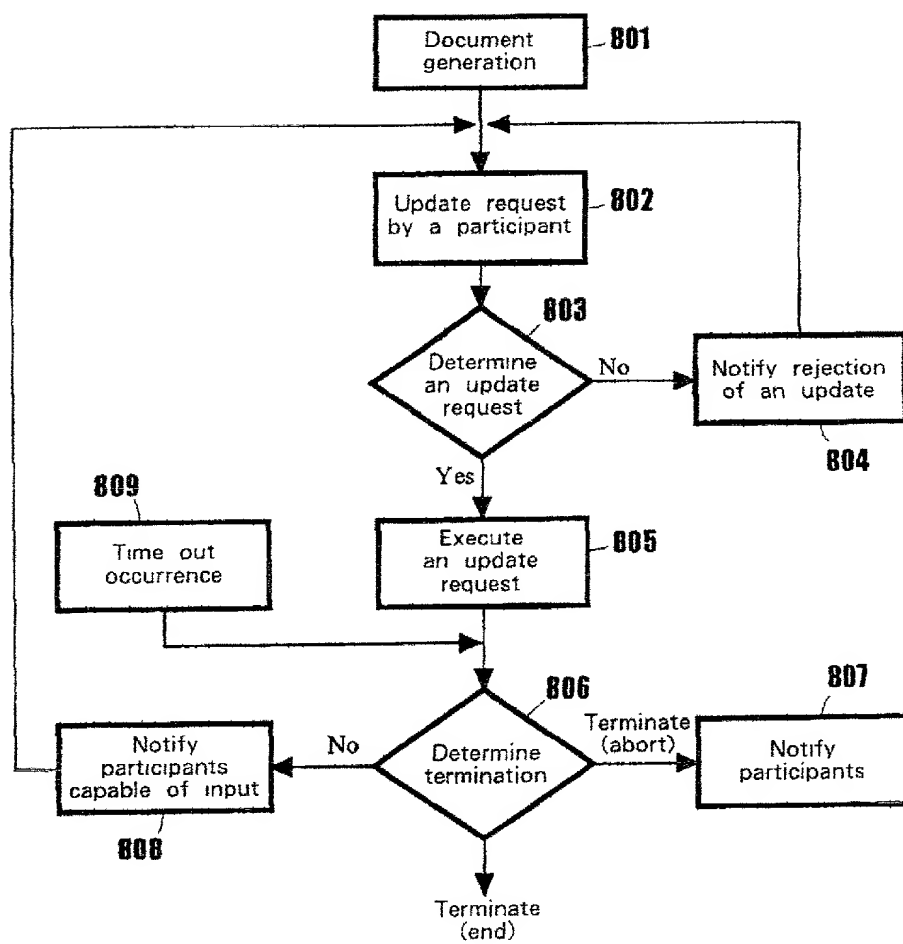
Document data structure

Fig. 6

<p>Contents</p> <p>OrderID= "00001"</p> <p>Consumer</p> <p> ConsumerID= "ConsumerA"</p> <p> Name= "Neyama"</p> <p> Address= "Yamato-shi"</p> <p> Phone= "042-123-4567"</p> <p> DeliveryDateRequested= "21/Sep/1999"</p> <p>Product</p> <p> ProductID= "IBM Aptiva"</p> <p> Price= "99,800 yen"</p> <p> UnitID= "9 116 54.89"</p> <p>Supplier</p> <p> SupplierID= "IBM Corp."</p> <p>Transport</p> <p> Specified= "Kuroneko"</p> <p> Candidate#0= "Pelican"</p> <p> DeliveryDateOffered= "21/Sep/1999"</p> <p> Candidate#1= "Kuroneko"</p> <p> DeliveryDateOffered= "20/Sep/1999"</p> <p> ⋮</p>	<p>History</p> <p>14/Sep/1999-15:20:30,Runtime,w,OrderID</p> <p>14/Sep/1999-15:22:20,Neyama,w,ConsumerID</p> <p>⋮</p> <p>14/Sep/1999-16:37:10,Pelican,c,Candidate#0</p> <p>14/Sep/1999-16:37:20,Pelican,w,Candidate#0</p> <p>Access Control</p> <p>value(ConsumerID),w,Specified</p> <p>Transport,c,Candidate#?,(value(Specified)-ml)</p> <p>Constrains</p> <p>value(DeliveryDateOffer) <= value(DeliveryDateRequested)</p> <p>timeout((isFilled(Specified)isFilled(DeliveryDateRequested)),100)</p> <p>Dependencies</p> <p>ConsumerID, OrderID</p> <p>Candidate#?, DeliveryDateRequested</p> <p>Termination</p> <p>End</p> <p> value(Specified) ! ml</p> <p>Abort</p> <p> ProductID, c,</p> <p> time(Specified, w) > time(DeliveryDateRequested) +</p>
---	--

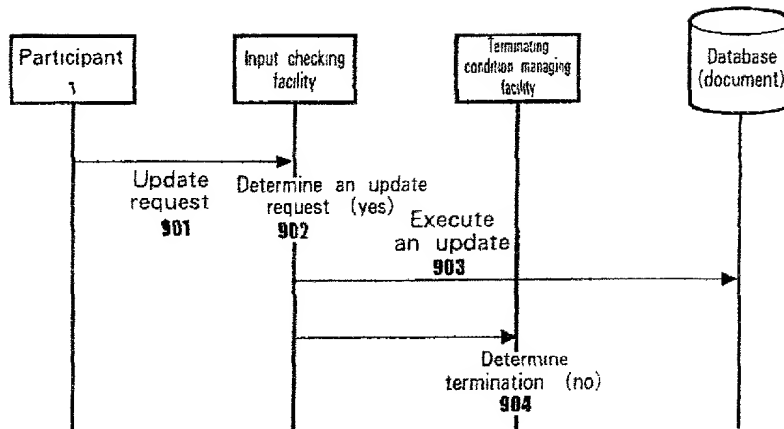
Example of a document

Fig. 7



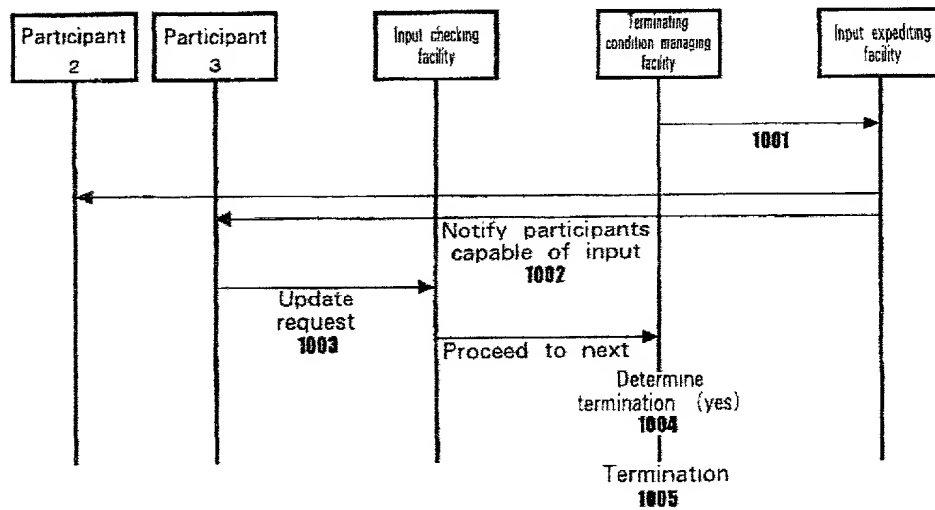
Operation of a flow control section

Fig. 8



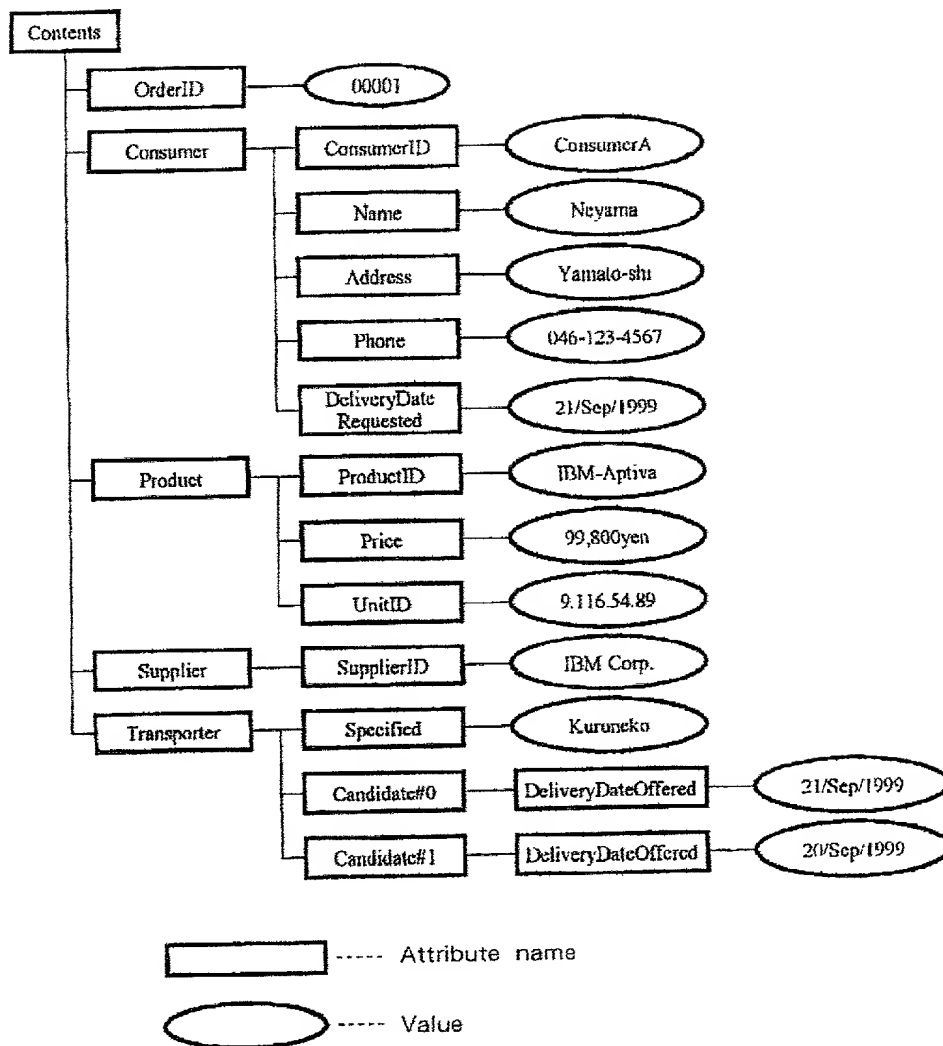
Processing flow among modules (1)

Fig. 9



Processing flow among modules (2)

Fig. 10

Structure of contents

	Node ID (Attribute name)	Parent node ID (Attribute name)	Value
T0	/	nil	nil
T1	/document	/	nil
T2	/document/contents	/document	nil
T3	/document/contents/OrderID	/document/contents	00001
T4	/document/contents/Consumer	/document/contents	nil
T5	/document/contents/Consumer /ConsumerID	/document/contents /Consumer	Neyama
T6	/document/contents/Consumer /ConsumerID/Name	/document/contents /Consumer	Ryoh Neyama
T7	/document/contents/Consumer /ConsumerID/Address	/document/contents /Consumer	Yamato-shi
T8	/document/contents/Consumer /ConsumerID/Phone	/document/contents /Consumer	046-123-4567

Representation of a tree structure of contents as a table

Order	Time (sec)	Writer ID	Action	Node ID
0	0	Runtime	Write	/document/contents/OrderID
1	100	Neyama	Write	/document/contents/Consumer /ConsumerID
2	100	Neyama	Write	/document/contents/Consumer /Name
3	100	Neyama	Write	/document/contents/Consumer /Address
4	100	Neyama	Write	/document/contents/Consumer /Phone

(Action types : Create, Write, Read, Cancel)

Example of History representation

Fig. 13

Outline part format
 allow(<node>,<user>,<operation>)

Example of rules
 Rule 1
 allow(?Node, ?User, "+w") ←
 isPath(?Node, "/document") and
 hasRole(?User, "Consumer").

Rule 2
 allow(?Node, ?User, "+w") ←
 isPath(?Node, "/ProductID") and
 hasRole(?User, "Consumer") and
 isCreator(?User, ?Node).

Example of Access Control representation

Fig. 14

Constraints 1

Contents : member(TransportSpecified, CompanyID)

Internal representation :

getValue("TransportsSpecified",V1) and

getValueList('CompanyID',V2) and

```
member( V1, V2 )
```

Constraints 2

Contents : DeliveryDateOffered <= DeliveryDateRequested

Internal representation :

getValue('DeliveryDateRequested',V1) and

getValue('DeliveryDateOffered',V2) and

$$V1 \leq V2$$

Example of Constraints representation

Fig. 15

Depended node ID	Dependent node ID
ProductID	UnitID
UnitID	TransportInfo
TransportInfo	TransportSpecified

Example of Dependencies representation

Fig. 16

```
(3) timeout(
    isSpecified( 'ProductID' ),
    isSpecified( 'TransportSpecified' ),
    180).
```

Fig. 17

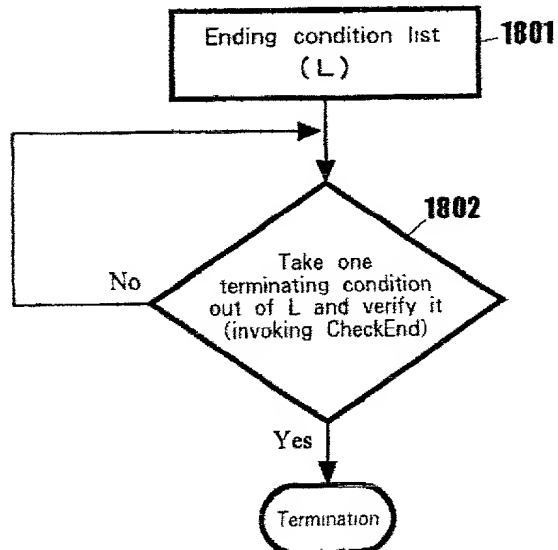
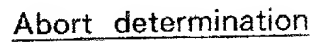


Fig. 18

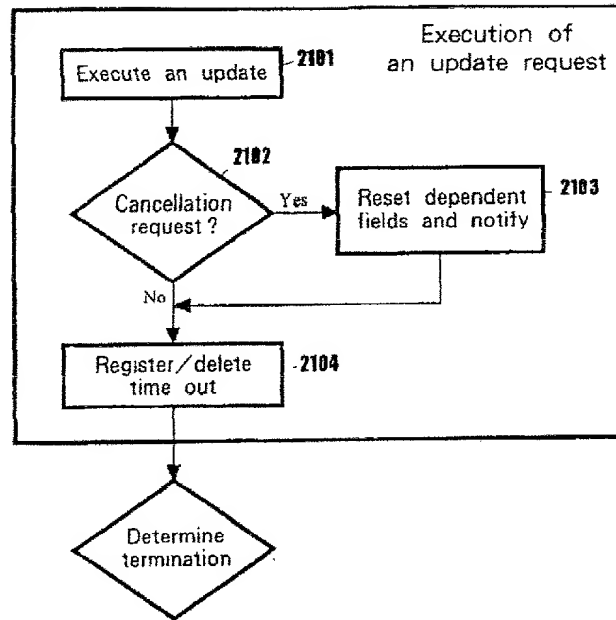


```

graph TD
    Start(( )) --> 2001[Input condition C]
    2001 --> 2002[Retrieve one data set DS meeting condition C]
    2002 --> 2003{Retrieval successful?}
    2003 -- No --> 2004[Output a list]
    2003 -- Yes --> 2005[Add DS to the list]
    2005 --> 2002

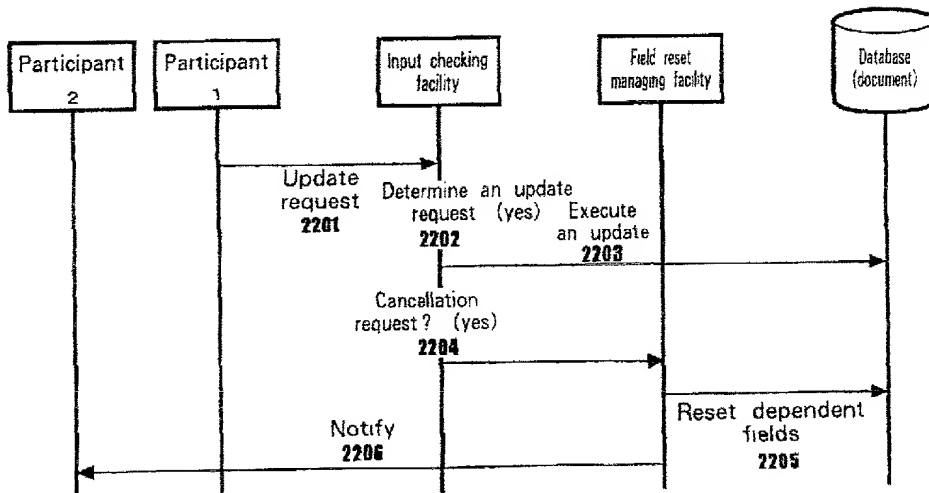
```

Fig. 20



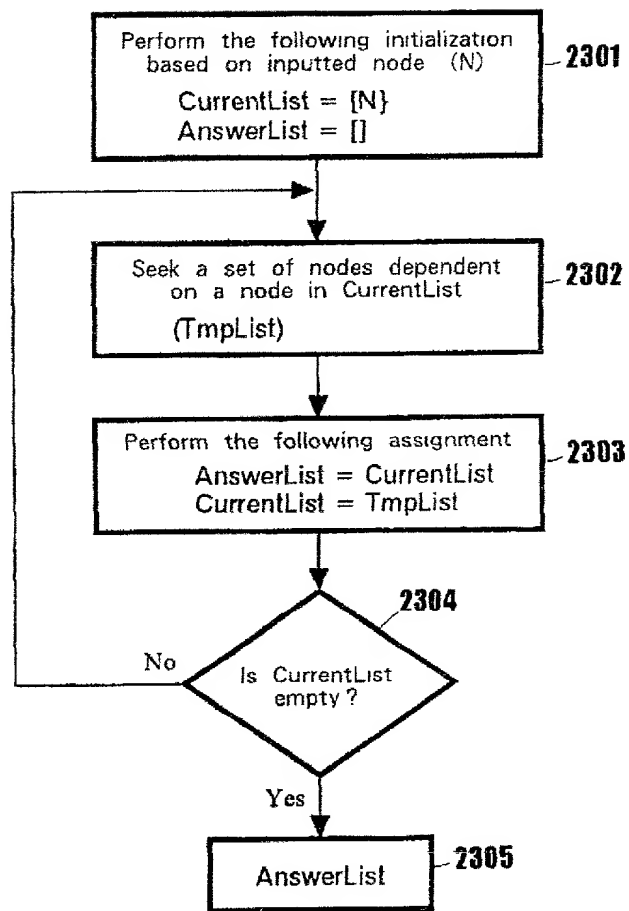
Details of execution of an update request

Fig. 21



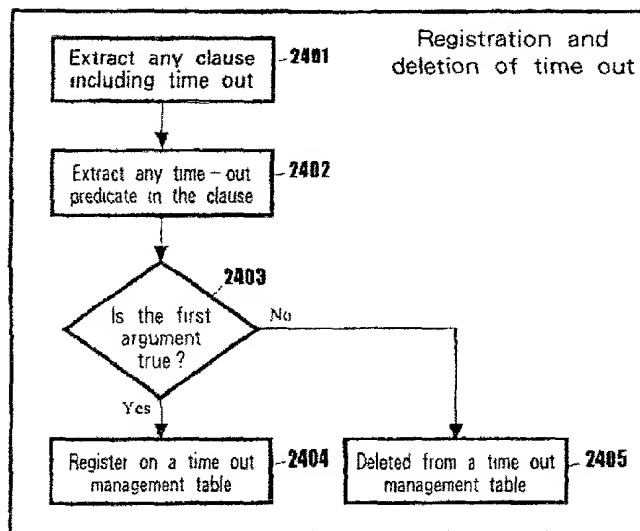
Details of update processing

Fig. 22



Processing for finding dependent nodes

Fig. 23



Details of registration and deletion of time out

Fig. 24

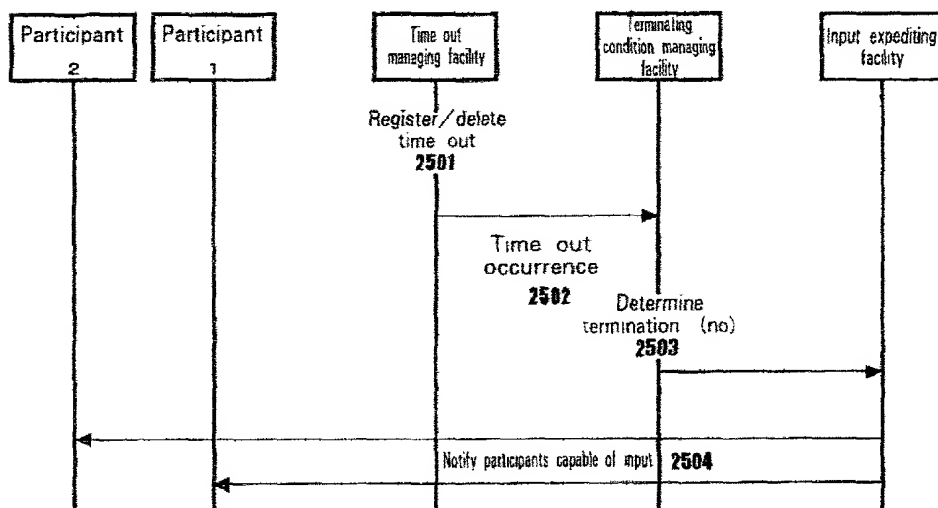


Fig. 25